



EPITA RENNES

RAPPORT DE SOUTENANCE FINALE

Bitume Legends



CARRENIX

Anthony Caron
Victorien Cambourian

9 Juin 2022

Melvyn Delaroque
Xavier de Place

Table des matières

Introduction	3
1 Origine du projet	3
2 Concept de base	3
3 Organisation interne chez CARRENIX	4
3.1 Organisation Pratique	4
3.2 Répartition des Tâches	4
3.2.1 Melvyn	5
3.2.2 Victorien	5
3.2.3 Anthony	6
3.2.4 Xavier	6
4 Récit chronologique	8
4.1 Janvier → Mars	8
4.1.1 Multijoueur	8
4.1.2 Programme Beta (β)	8
4.1.3 Communication	9
4.1.4 Site Internet	9
4.1.5 Menu	9
4.2 Mars → Avril	10
4.2.1 Graphisme	10
4.2.2 Sauvegarde	11
4.2.3 Menu	11
4.2.4 Intelligence Artificielle	11
4.2.5 Gameplay	12
4.2.6 Musiques	13
4.2.7 Site Internet	13
4.3 Avril → Juin	13
4.3.1 Sauvegarde	14
4.3.2 Menu	14
4.3.3 Multijoueur	14
4.3.4 Intelligence Artificielle	14
4.3.5 Gameplay	15
4.3.6 Sound Design et Musiques	16
4.3.7 Site Internet	16

5	Partie Technique	16
5.1	Collaboration	16
5.2	Communication	17
5.3	3D	17
5.4	IDE	17
5.5	Installateur	18
6	Notre ressenti au long du projet	18
6.1	Janvier → Mars	18
6.1.1	Anthony	18
6.1.2	Melvyn	19
6.1.3	Victorien	19
6.1.4	Xavier	19
6.2	Mars → Avril	20
6.2.1	Anthony	20
6.2.2	Melvyn	20
6.2.3	Victorien	20
6.2.4	Xavier	21
6.3	Avril → Juin	21
6.3.1	Anthony	21
6.3.2	Melvyn	21
6.3.3	Victorien	21
6.3.4	Xavier	22
7	Ressentis de certains testeurs	22
8	Problèmes rencontrés et solutions	23
8.1	Janvier → Avril	23
8.1.1	Implémentation des voitures	23
8.1.2	Intelligence Artificielle	23
8.1.3	Implémentation des musiques	24
8.2	Avril → Juin	24
8.2.1	Implémentation des voitures	24
8.2.2	Implémentation des musiques	25
8.2.3	Intelligence Artificielle	26
8.2.4	Multijoueur	26
Conclusion		27
8.3	Objectif atteint ?	27
8.4	Avenir de Bitume Legends	27
8.5	Remerciements	28

Introduction

Nous sommes le Studio CARRENIX et nous sommes fiers de vous présenter le rapport de notre projet, *Bitume Legends*. Dans ce document, vous découvrirez comment nous avons mené à bien notre projet, mais également comment de fil en aiguille nous avons abouti à ce résultat. Suivant un plan chronologique, nous aborderons les différents mode de jeu, notre façon de travailler, notre approche du jeu, mais également sa promotion et ce que nous avons tiré des différentes soutenances ainsi que ce que nous a apporté ce projet. Nous concluons avec des remerciements aux différentes personnes qui ont contribué à faire avancer le projet *Bitume Legends*.

1 Origine du projet

Dans notre groupe, nous aimons beaucoup les voitures. Nous regardons tous des émissions comme *Top Gear*, *The Grand Tour* ou encore *Vilebrequin*. Nous avons donc décidé de faire un projet qui se rapporte à cet univers. Nous avons réfléchi à plusieurs possibilités de jeux, mais nous nous sommes mis d'accord sur un jeu de courses plutôt qu'un RPG où les joueurs incarneraient des voitures. De plus, un jeu de courses est facilement imaginable avec un mode multijoueur et un mode Intelligence Artificielle. C'était donc un choix qui s'imposait pour nous.

Le nom *Bitume Legends* est une référence au jeu de Gameloft, *Asphalt 9 Legends*. Nous connaissons tous ce jeu et nous avons décidé de nous inspirer grandement de son nom tout en y intégrant une de nos spécialités : les jeux de mots. Le nom de notre team CARRENIX est du même type. Nous avons francisé le nom du studio de jeux vidéo Square Enix. En français, cela donne Carré Enix, réduit en CARRENIX.

2 Concept de base

Notre vision originale était de faire un jeu de course automobile en style *Low Poly*. Nous voulions avoir un système d'expérience, permettant de débloquent de nouvelles voitures plus puissantes. Pour acquérir cette expérience, nous voulions que le joueur puisse faire des courses de plusieurs types. Le premier serait de se battre contre la montre, c'est à dire faire des courses en temps limité. Le second serait de se battre contre une Intelligence Artificielle, de faire la course contre une voiture automatique. Et enfin, nous voulions avoir une option multijoueur, pour que nous puissions faire des soirées avec nos amis et jouer à plusieurs à notre

jeu. Pour que le jeu ne soit pas ennuyant, nous voulions créer différents circuits, dans différents endroits et avec des atmosphères différentes. A cela, nous voulions ajouter des voitures aux caractéristiques réalistes, et aux designs variés. Enfin, nous voulions avoir notre propre bande son, composée par nos soins, et notre charte graphique.

3 Organisation interne chez CARRENIX

3.1 Organisation Pratique

Nous avons mis en place une organisation particulière entre nous, car au début nous ne savions pas spécialement par où commencer. Chaque semaine, nous nous sommes réunis pour définir des objectifs pour chacun, à faire durant la semaine suivante. Cela nous a permis d'avoir des buts concrets sur le court terme et d'avancer plus efficacement.

3.2 Répartition des Tâches

Nous nous sommes basés sur ce que nous avons annoncé dans notre Cahier des Charges : Melvyn s'est occupé des musiques d'ambiance, du *Sound Design* de notre jeu et du menu principal. Victorien a créé le premier circuit du jeu, il a implémenté le menu principal et a modélisé sur *Blender* une Formule 1. Anthony a géré le site Internet, les réseaux sociaux et le programme β . Enfin, Xavier a implémenté le mode multijoueur, et le cœur du moteur de jeu, à savoir le système de direction des voitures.

Annexe : Tableau récapitulatif du projet

Tâches	Période 1	Période 2	Période 3	Responsable	Suppléant
Multijoueur	100%	100%	100%	Xavier	Victorien
Menu principal	85%	95%	100%	Melvyn	Xavier
Logique du jeu	100%	100%	100%	Victorien	Anthony
Programme β	50%	75%	100%	Anthony	Melvyn
Site Internet	50%	85%	100%	Anthony	Victorien
Maps et niveaux	0%	95%	100%	Victorien	Melvyn
Intelligence Artificielle	0%	100%	100%	Xavier	Anthony
Implémentations modes de jeu	0%	100%	100%	Victorien	Xavier
Systèmes de gains	0%	10%	100%	Anthony	Xavier
<i>Sound Design</i>	10%	45%	100%	Melvyn	Victorien
Musiques	10%	45%	100%	Melvyn	Anthony

TABLE 1 – Tableau récapitulatif de notre projet

3.2.1 Melvyn

Étant responsable de la musique, j’ai composé la musique du jeu. J’ai utilisé le logiciel de MAO (Musique Assistée par Ordinateur) *FL Studio 20*. J’ai composé deux musiques, une musique pour le menu et une musique de course et j’ai demandé à un ami d’enregistrer sa voix pour la musique dans un studio d’enregistrement de son école. Le style choisi par mes pairs et moi-même au sein du groupe est la *Phonk*, style est souvent associé à la culture automobile *underground* japonaise, d’où le désir d’utiliser de la *Phonk* pour notre jeu. De plus, ce style est très énergique, assourdissant parfois et fait monter l’adrénaline dans le sang. Au début, j’ai hésité à faire de l’*Eurobeat*, un genre de *dance music* associé au milieu automobile, mais le groupe a tranché et nous avons choisi un genre qui nous est à la fois plus familier et bien plus facile à produire comparé à de la musique électronique. De plus je suis responsable du design et j’ai travaillé sur le design des menus, ainsi que sur l’identité graphique du jeu et l’identité sonore de *Bitume Legends*.

3.2.2 Victorien

Étant donné que nous créons un jeu de voiture, il était bienvenu de modéliser au moins une voiture par nos propres soins. Ayant des connaissances dans *Blender*,

un logiciel de modélisation 3D, je m'en suis chargé. Nous avons convenu d'une Formule 1, voiture considéré comme le "Graal" du jeux. Les formes de la voiture ont toutes été créées à partir de formes basiques, telles que des rectangles, des cylindres, des sphères ou encore des cubes. Ensuite, il ne restait qu'à jouer avec les formes : les étirer, redimensionner une face pour que les formes correspondent avec une Formule 1. Vu que le design est en *low-poly*, nous avons rajouter des faces à certaines formes. Par exemple, avec 26 rectangles verticaux collés les uns aux autres, on peut former un cylindre. Pour essayer d'être réaliste, nous avons regarder une vidéo présentant une Formule 1 avec ses détails telles que le volant, les suspensions, les ailerons et les moustaches de la voiture.

3.2.3 Anthony

Pour le site Web, nous avons décidé de reprendre l'esthétique du menu de notre jeu vidéo. Pour cela, nous avons divisé le site web en quatre parties, accessibles par quatre boutons amenant aux quatre sous-pages de notre site. La première est la présentation du projet, dans laquelle nous avons écrit une courte introduction, l'historique et les membres de l'équipe. La seconde concerne la réalisation du projet, qui contient la chronologie de la réalisation, les problèmes rencontrés et solutions envisagées pour contrer ceux-ci. Ensuite, une troisième page nous offre des liens de téléchargement du jeu pour Windows et macOS, et le projet en version complète ou allégée. Enfin, la dernière page permet l'accès aux différentes ressources (liens des images, sons, logiciels...). C'est une sorte de bibliothèque des sources de notre projet. Ainsi, il me manquera plus qu'à faire fonctionner certains boutons et liens et rendre le site responsive, c'est à dire un site qui s'adapte en fonction de l'appareil depuis lequel on le consulte, et régler quelques bugs d'affichages pour finaliser le site Web.

3.2.4 Xavier

Pour faire tourner un jeu, il faut que chaque élément soit relié avec les autres grâce à des liens dans le code. Dans notre cas, cela peut être appliqué aux voitures, qui doivent être des objets controllables et visbles. Pour cela, nous avons intégré les voitures dans le moteur de jeu *Unity* en rajoutant des composants tel que le *Rigidbody*, le centre de gravité, les *Colliders*, et les *Wheel Colliders*. Le premier est un élément permettant de simuler les propriétés physiques de la voiture, comme la masse ou la détection de collisions. Le second est un élément du *Rigidbody*, qui est sous la forme d'un point précis mis sur la voiture, et qui est le point d'application de la gravité. C'est un élément pratique, qui permet de définir comment la voiture va réagir dans les virages, ou comment celle-ci va *drifter*. Le troisième élément à intégrer est en plusieurs parties, chacune pour un morceau de la voiture (capot,

toit, ailes, etc.). Ce sont des zones virtuelles autour de la voiture qui permettent de la rendre solide, c'est à dire qu'on ne peut pas traverser, et qui ne peut pas traverser non plus les murs et autres éléments. Et enfin, les *Wheel Colliders* sont des éléments appliqués sur chaque roue et qui jouent le rôle de moteur. Pour les roues arrières, ils transmettent l'action du joueur si il appuie sur les touches pour avancer et reculer, en entrainant les roues (et donc la voiture) vers la bonne direction. Et pour les roues avant, ils s'occupent de faire tourner la voiture en fonction des actions du joueur. Après cela, nous avons intégré le mode de jeu mutlijeueur. Nous avons décidé d'utiliser *Photon Cloud*, une solution gratuite, *open source*, et relativement "simple" à implémenter. Il y a une bonne documentation sur Internet, et de nombreux tutoriaux sur YouTube. Nous avons commencé par créer une scène pour la connection, qui, grâce à une interface graphique, permet de choisir un nom. Ensuite, nous pouvons créer une *room* (instance de multijoueur), d'en chercher une déjà ouverte ou encore d'en joindre une directement si nous connaissons son nom. Enfin, une fois que la *room* est créée, le jeu lance une scène comportant un circuit, et le jeu peut commencer.

4 Récit chronologique

DANS CETTE PARTIE, NOUS VOUS PRÉSENTONS COMMENT NOUS NOUS SOMMES ORGANISÉS POUR SORTIR NOTRE JEU BITUME LEGENDS, MAIS AUSSI COMMENT CHAQUE PARTIES DU JEU ONT ÉTÉ RÉALISÉES

4.1 Janvier → Mars

Période 1 : Remise du cahier des charges \implies Première soutenance

1. Implémentation du mode Multijoueur
2. Création du menu principal du jeu
3. Implémentation de la logique basique du jeu (mouvements des voitures, gestion des crash, etc.)
4. Création du programme de β -testing

4.1.1 Multijoueur

Nous avons implémenté la base du multijoueur, non sans peines. Nous avons eu des problèmes de synchronisation entre les joueurs mais également des problèmes de vues et de contrôle. C'est à dire que nous contrôlions la mauvaise voiture et que nous ne voyions pas par notre caméra mais celle d'un autre joueur. Nous avons choisi de limiter la connexion à 4 personnes en simultané, pour éviter les "embouteillages" dans le circuit tout en offrant un minimum de compétition. Il y a une interface de connexion, avec différentes options (détaillées dans le manuel d'utilisation), nous permettant de choisir un pseudo, de créer ou rejoindre une instance du jeu, et de commencer le jeu.

4.1.2 Programme Beta (β)

Nous avons créé un questionnaire sur *Google Forms* afin que les premières personnes testant le jeu puisse nous signaler les bugs qu'elles ont rencontrés et nous les décrire le plus précisément possible, dans l'optique de les corriger rapidement. Pour cela, ce questionnaire possède plusieurs questions nous permettant de cibler le plus précisément possible le problème tels qu'un problème d'interface, un mode de jeu disfonctionnel mais encore un potentiel cheater. Toutefois, l'utilisateur peut nous contacter sur *Discord* ou *GitHub* si besoin.

4.1.3 Communication

Nous avons créé un compte *Instagram*¹ pour tenir informé nos potentiels premiers joueurs. Ils pourront nous rejoindre sur *Discord* pour que nous puissions les inscrire dans le programme de tests. Nous avons également utilisé notre réseau d'amis mais également les différentes présentation de ce que nous faisons dans notre lycée pour mettre en avant notre projet tout en leur montrant un projet concret qui peut donner envie.

4.1.4 Site Internet

Le site Internet a pour but de présenter les membres du groupe, leur rôle et leur avis sur le projet. Il présente aussi l'historique et les idées de conception du jeu, du nom du jeu et celui du studio de développement. Ensuite, il nous permet d'expliquer comment nous avons réalisé le projet avec la chronologie de réalisation, c'est-à-dire les différentes étapes de réalisation du projet mais aussi les problèmes rencontrés et les solutions envisagées. Enfin, il contient aussi un espace dédié pour télécharger le projet et le jeu. Ce site est notre vitrine, il nous permet de faire la promotion de *Bitume Legends*.

4.1.5 Menu

Le menu doit permettre d'accéder aux différents modes de jeux et fonctionnalités du jeu. Il doit servir de hub pour accéder au mode multijoueur, contre la montre et contre l'Intelligence Artificielle. Il doit également permettre d'accéder au garage contenant les voitures du joueur. Un affichage du niveau du joueur, un bouton pour accéder à la remontée de bug et un menu d'options sont également dans le menu du jeu. Au début du projet, le menu était basique. Il servait simplement à pouvoir accéder aux différentes modes de jeux. Suite à cela, Melvyn et Victorien ont commencé à travailler sur un design plus poussé que celui de base de *Unity*, en adoptant notamment un thème de couleurs basé sur le logo du jeu. Tout d'abord, nous avons réalisé une ébauche du design sur Canva, dans le but de partager des idées sur le look final du menu. Ensuite nous avons implémenté le design dans le jeu, en insérant un fond et des espaces pour les futures fonctionnalités, en implémentant les boutons et en changeant leur design. Nous avons également implémenté la charte graphique du jeu. Enfin, nous avons ajouté des effets sonores et visuels aux boutons lorsqu'ils sont cliqués ou lorsque la souris passe dessus. Une musique composée par Melvyn tourne également en fond dans le menu.

1. [instagram.com/bitumelegends](https://www.instagram.com/bitumelegends)

4.2 Mars → Avril

Période 2 : Première soutenance \implies Seconde soutenance

1. Création des *maps* de jeu en fonction des niveaux
2. Système de sauvegarde
3. Implémentation des différents modes de jeu (contre l'Intelligence Artificielle, contre la montre, etc.)
4. Implémentation de l'Intelligence Artificielle
5. Amélioration du Gameplay
6. Correction des bugs
7. Création et mise en ligne du site Internet

Suite à la soutenance 1 du 10 mars, nous avons pris quelques jours de repos. Puis le 14, nous avons mis en place nos objectifs pour la soutenance 2 vis-à-vis du cahier des charges, de l'état du jeu ainsi que des commentaires de la première soutenance.

Nous avons travaillé en parallèle sur différents objectifs, certains plus longs que d'autres. Notamment l'Intelligence Artificielle, qui a prit énormément de temps et son travail d'implémentation ayant commencé au début de cette période. Le 20 mars ce sont les mécaniques de conduite comme le drift ainsi que le *sound design* des véhicules qui ont été implémentés. Puis le 25 la création de deux nouveaux circuits, *City* et *Port*. Le mois de mars s'est conclu par la création des menus des modes de jeu : *Timer* et *Solo*.

Après 3 semaines, la première version de l'Intelligence Artificielle était prête le 10 avril, ainsi que le comportement en fin et début de course le 15. Le 18, le garage et le choix de la voiture sélectionnée, ainsi que la sélection de la difficulté et du circuit ont été mis en place avec succès avec le système de sauvegarde. Le 20, l'IA a connu sa version 2.0 avec une précision améliorée et fonctionnelle sur les deux circuits.

Durant la dernière semaine avant la soutenance, des implémentations mineures ou des corrections de bugs ont eu lieu. Le 22 avril le menu principal a été terminé et avec l'affichage de la voiture et le lien entre chaque sous-menus a été fait. Le 24 la musique a été correctement implémentée et le 25 le mode de jeu *Timer* a été corrigé pour enfin fonctionner correctement. Enfin, le 27 le site Internet est devenu *responsive* et le 28 le mode *Solo* est devenu fonctionnel.

4.2.1 Graphisme

Au lancement du projet, nous voulions créer nous même nos voitures et nos circuit à l'aide du logiciel de modélisation 3D Blender. Toutefois, cela prends beau-

coup de temps, temps qui n'est pas utilisé à coder le jeu, résoudre des bugs, etc. Nous avons donc fait un compromis entre notre volonté de créer nous même le design du jeu et la contrainte de temps. Nous nous sommes orientés alors sur l'ajout d'un *asset* du *Unity Asset Store*, pour obtenir des voitures et bases de circuits. Ceci nous a permis de créer directement dans *Unity* nos différents circuits et de gagner beaucoup de temps.

4.2.2 Sauvegarde

Pour que notre jeu ne soit pas une prise de tête à chaque lancement et devoir tout recommencer, nous avons créé un système de sauvegarde. Il est composé de 2 principes : le *Save*, qui permet d'enregistrer les données voulues et nécessaires au bon fonctionnement du jeu, et le *Load*, qui permet de charger la sauvegarde lors du lancement du jeu. Ce système permet ainsi de sauvegarder la voiture sélectionnée et de la charger directement dans le menu. Il permet également de sauvegarder les paramètres sonores du jeu (volume de la musique) mais également l'expérience du joueur, et ses voitures débloquées. À tout moment, le joueur peut réinitialiser sa progression et recommencer le jeu depuis le début.

4.2.3 Menu

Pour la première soutenance, nous avons fait une charte graphique et un design pour les menus. Nous nous étions concentrés sur le menu principal. Pour la seconde soutenance, nous avons travaillé les sous-menus des différents modes de jeu et les fonctionnalités internes.

Nous avons implémenté le garage, permettant de sélectionner une voiture pour la course parmi la liste de voitures disponibles, qui est ensuite sauvegardée. Nous avons ensuite modifié le menu de telle façon que La voiture sélectionnée s'affiche ensuite dans le menu principal à place du logo du jeu. Les menus des modes de jeux *Timer* (course contre la montre) et *Solo* (contre une Intelligence Artificielle) ont été implémentés de la même manière. Ils ont donc une structure similaire. Au sein de ces menus, nous retrouvons la sélection des circuits ainsi que la difficulté de la course et un moyen de revenir au menu précédent.

Nous comptons ajouter un menu de réglages permettant de régler le volume du jeu, le volume des musiques, les touches permettant d'avancer ainsi que de changer de pseudo. Une version simplifiée de ce menu, contenant la modification des touches de directions et le volume du jeu sera également disponible en course.

4.2.4 Intelligence Artificielle

L'implémentation de l'Intelligence Artificielle s'est bien déroulée. Nous avons pris du temps à nous décider sur quelle solution nous allons utiliser et sur com-

ment l'Intelligence Artificielle devrait se comporter une fois implémentée dans le jeu. Nous sommes partis sur une solution hybride entre celle intégrée dans *Unity* et une *homemade*. Nous nous sommes basé sur le *NavMesh* de *Unity*, puis nous avons travaillé à animer la voiture et à définir les points qu'elle devait franchir pour terminer le circuit. Ensuite est arrivée la (longue) partie de la calibration. Au début, notre IA se déplaçait aléatoirement, si bien qu'elle inventait le chemin à chaque fois sans prendre le circuit que nous avions dessiné. Après plusieurs jours de recherche, nous avons réussi à lui faire prendre uniquement le chemin prévu. Ensuite, il a fallu régler sa vitesse et sa précision, pour éviter qu'elle ne rentre dans chaque mur par souci de freinage en virage.

Après ces quelques soucis, nous avons obtenu un mode *Solo* pratiquement fonctionnel. Nous avons rajouté à ceci les scripts qui nous permettent de gérer le départ et la fin de la course et nous étions bons. Il est à noter que l'Intelligence Artificielle possède la même physique que les autres voitures présentent dans le jeu.

4.2.5 Gameplay

En début de course, un décompte avant le départ est donné, bloquant la voiture pour empêcher les faux départs. Tout au long du circuit, le joueur doit traverser une série de balises, les *checkpoints*, pour valider la course. Cela permet d'empêcher le joueur de simplement faire demi-tour et de traverser la ligne d'arrivée pour gagner, ou de couper le circuit. Ici le joueur est forcé de passer par chaque *checkpoints* dans le bon sens afin de pouvoir terminer la course, idem pour l'Intelligence Artificielledans le mode *Solo*.

Chaque mode de jeu (hors multijoueur) comprend une sélection de difficulté. La difficulté pour le mode de jeu *Timer* est déterminée par le temps maximum pour terminer la course. En revanche, pour le mode de jeu *Solo*, elle réside dans la vitesse et la précision de l'Intelligence Artificielle, ce qui permet d'affronter des *bots* plus ou moins forts.

En mode *Solo*, la course est gagnée si l'on est arrivé en premier ou que l'on a survécu au *bot* qui peut nous infliger des dégâts. En mode *Timer*, elle est gagnée lorsque l'on passe la ligne d'arrivée (et toutes les balises précédentes) avant la fin du temps imparti. En cas de défaite, pour chacun de ces deux modes de jeu la partie est terminée et l'on peut soit recommencer la course, soit revenir au menu pour changer de véhicule, de circuit ou de difficulté.

Nous avons également mis en place un système de collision qui, en plus d'impacter la forme de la voiture, peut aussi impacter son comportement (un mauvais choc sur une des roues avant ou le coté de la voiture peut provoquer des soucis de direction).

4.2.6 Musiques

Notre jeu comporte des musiques originales composées sur *FL Studio 20* par Melvyn. Un objectif d'un peu moins de 5 musiques différentes a été posé dans le cahier des charges. Suite aux commentaires de la première soutenance où il nous a été demandé de plus se concentrer sur le jeu, seules une musique de menu et une musique de course ont été composées. Toutefois, nous tenions à ce qu'il y ai des nouveautés sur l'aspect de l'ambiance du jeu. Alors un travail sur l'implémentation de ces musiques a été faite, notamment lors du passage d'une scène à une autre sans que la musique ne cesse ou recommence depuis le début. Pour la troisième soutenance, l'objectif des musiques sera rempli avec un système de sélection des musiques avec la possibilité de modifier le volume de la musique dans un sous-menu de réglages comme dit ci-dessus dans la section menu.

4.2.7 Site Internet

Pour ce qui est du site Internet, la mise en page de ce dernier était déjà bien avancée. Cependant, quelques ajouts ont été fait, notamment pour permettre la finalisation de la page ressources, qui contient tous les outils que nous avons utilisé pour réaliser notre jeu. De plus, la page réalisation a été terminée avec l'ajout de la partie problèmes et solutions. Enfin, pour ce qui est de la partie la plus difficile du site Internet, le *responsive design*, (qui est de rendre le site Internet adaptable à toutes tailles d'écran), l'utilisation du code HTML était indispensable. Nous avons donc cherché à comprendre comment le *responsive* fonctionnait à l'aide de tutoriels et implémenté chaque élément pour les rendre indépendants entre eux et ainsi obtenir notre site Internet *responsive*.

4.3 Avril → Juin

Période 3 : Seconde soutenance \implies Soutenance finale

1. Finalisation des niveaux et des systèmes de gains
2. Mise en place des musiques et du *sound design* (menu, en course, multijoueur)
3. Finalisation du mode Multijoueur
4. Refonte de l'intelligence artificielle
5. Finalisation du site Internet
6. Correction des bugs

4.3.1 Sauvegarde

Le système de sauvegarde est indispensable pour déterminer l'avancer d'un joueur dans le jeu. C'est pourquoi nous avons décidé de mettre en place une manière de représenter cette avancement par un système de gain et de niveau, qui est sauvegardé dans le jeu. L'utilisateur acquiert de l'expérience plus ou moins vite en fonction de son niveau. Un joueur finissant souvent premier en multijoueur, battant l'IA et gagnant contre la montre en mode difficile se verra gagner beaucoup plus d'expérience qu'un joueur arrivant souvent dernier en multijoueur mais aussi un autre, gagnant contre la montre en mode facile. Toute cette expérience est nécessaire pour passer d'un niveau à l'autre. Un joueur démarre le jeu au niveau 0 et grimpe les échelons de plus en plus difficilement (puisqu'il faut plus d'expérience entre chaque niveau) jusqu'au niveau maximum, le niveau 4. A chaque niveau, le joueur débloque une nouvelle voiture pour le récompenser de son activité dans le jeu.

4.3.2 Menu

Après avoir implémenté le système de gain et de niveau, il nous faut maintenant un moyen de représenter l'expérience et le niveau actuel du joueur. Pour cela, une barre représentant l'expérience du joueur est afficher dans le menu ainsi que son niveau. Bien évidemment, ces deux éléments sont mis à jour après chaque partie du joueur. De plus, le bouton "Option" a été rajouté pour régler le son et la musique du jeu, mais aussi pour permettre à l'utilisateur de choisir sous quelle résolution il veut jouer et avec quelle qualité de texture.

4.3.3 Multijoueur

Sed eleifend, eros sit amet faucibus elementum, urna sapien consetetuer mauris, quis egestas leo justo non risus. Morbi non felis ac libero vulputate fringilla. Mauris libero eros, lacinia non, sodales quis, dapibus porttitor, pede. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Morbi dapibus mauris condimentum nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Etiam sit amet erat. Nulla varius. Etiam tincidunt dui vitae turpis. Donec leo. Morbi vulputate convallis est. Integer aliquet. Pellentesque aliquet sodales urna.

4.3.4 Intelligence Artificielle

Nous avons totalement refait notre système d'Intelligence Artificielle, car la précédente n'était pas vraiment fonctionnelle. Auparavant l'ancienne IA "glissait sur

le sol" sans utiliser les contrôles du véhicule. Désormais le nouveau système d'Intelligence Artificielle prend en compte le système de physique du jeu. Ce nouveau système fonctionne par l'utilisation d'une série de *Waypoints* et de *Breakzones*. Ces *Waypoints* sont des zones circulaires d'une certaine taille que la voiture doit traverser dans l'ordre. Elle établit un chemin le plus court en fonction du placement des points et de son placement et le calcule en prenant en compte les obstacles sur le chemin. Le calcul se fait en direct afin de lui permettre de retracer un nouvel itinéraire jusqu'au prochain *Waypoints* dans le cas où elle aurait été déviée de sa course, par un joueur plus particulièrement. Les *Breakzones* sont des zones de taille modifiable qui forcera le bot à atteindre une vitesse définie au préalable afin d'anticiper des virages. Ce système permet à la voiture d'avoir le meilleur comportement possible et laisse beaucoup de place à l'optimisation, afin d'avoir le meilleur parcours possible. Sur chaque circuit nous comptons 3 Intelligence Artificielles, une pour chaque mode de difficulté. Elles se différencient par le véhicule utilisé, plus on augmente en difficulté, plus le véhicule sera sportif, nerveux et rapide. Bien que toutes les IA malgré la difficulté aient les mêmes *Waypoints* et *Breakzones*, ils ont été optimisés afin de pouvoir fonctionner pour tous les bots, et même d'être compétitifs.

4.3.5 Gameplay

Pour avoir un gameplay plaisant et attirant, nous avons ajoutés des objectifs ainsi qu'une interface simpliste, mais qui permet de donner un minimum de sensation au joueur. Dans le jeu, vous aurez la possibilité de mettre pause, et ainsi de relancer le jeu ou alors de quitter et revenir au menu. Une fois la partie terminée, vous pourrez encore une fois, soit relancer, soit quitter. Nous avons ajouté un compteur de vitesse, ainsi qu'un décor assez dense, de telle manière à ce que le joueur ressente un certain dynamisme ainsi qu'une sensation de vitesse. Nous avons également adapté les circuits de façon à ce que les joueurs ne soient pas perdus dans le circuit et sachent où aller instinctivement. Pour ce qui est des objectifs, il y a la volonté de gagner la course, mais également le déblocage de voitures dans votre garage. Vous commencez donc niveau 0, et si vous voulez jouer avec l'une des voitures les plus puissantes du jeu, vous devez gagner plusieurs courses, ce qui vous permet de connaître mieux les différents circuits, mais également les voitures que vous conduisez.

Suite aux remarques de la deuxième soutenance, nous nous sommes rendus compte du fait que nous avions un trop grand nombre de véhicules. Bien que la diversité de choix nous tennaient à cœur, la grande majorité des voitures n'avait rien de particulier à offrir, nous avons simplement réutilisé des paramètres fonctionnels à toutes les voitures sans trop nous y attarder. Cependant afin d'avoir un jeu misant sur la qualité et non pas sur la quantité, nous avons opté pour 5 véhicules différents,

qui cette fois-ci possèdent des caractéristiques leur étant propres et correspondant au type de véhicule. Chaque véhicule a des performances différentes, plus sportives à hauts niveaux.

4.3.6 Sound Design et Musiques

A l'origine on prévoyait une demi-douzaine de musiques pour le jeu, plusieurs en menu et plusieurs en course. Cependant le temps dédié à la composition musicale s'est avéré trop grand comparé au temps dédié à la création du jeu. Melvyn a donc décidé de ne pas en composer d'autre et s'est concentré sur le sound-design du jeu. Un des objectifs pour la troisième soutenance était également d'avoir un son caractéristique pour chaque véhicule car comme en vrai, en fonction du véhicule et de ses caractéristiques, le son émis est différents. Il a fallu récupérer les sons de chaque modèle utilisé, les traiter et gérer le comportement en volume et en hauteur en fonction du nombre de t/min. Les voitures modifiées peuvent également avoir un bruit de turbocompresseur ou de supercompresseur.

4.3.7 Site Internet

Au sujet du site internet, il ne restait pas grand chose à faire puisqu'il était déjà pratiquement finalisé. Nous avons donc terminé ce site en ajustant le mieux possible les éléments entre eux afin d'améliorer le visuel. De plus, quelques éléments ont été rajouté pour le rendre le plus complet possible. Enfin, nous avons pris le temps de relire attentivement le site dans le but de corriger tous les fautes que nous aurions pu voir.

5 Partie Technique

Pour la réalisation de ce projet, certaines étapes étaient incontournables. C'est pourquoi nous allons vous les expliquer en passant par les nombreux outils que nous avons utilisé et en quoi ils nous ont servi.

5.1 Collaboration

La création de jeux en solo ne demande pas de partage de données, contrairement à celle en équipe. Nous avons besoin d'un système de collaboration et de partage des données où nous pourrions nous échanger les fichiers relatifs au jeu. Nous avons donc créé un ensemble de *repositories* (ou *repo*) sur *Github*², répartis dans une organisation, chacun pour un usage bien spécifique. Le premier est

2. github.com/Bitume-Legends-Crew

donc le *repo* de notre jeu, nommé *game*, qui contient le projet au format *.unity* et des dossiers contenant les différents assets et autres ressources, nécessaires à l'exécution du jeu. Le second est un *repo* consacré exclusivement aux différents rapports ou au cahier des charges que nous devons fournir. Il est composé à 99% de *.TeX* et 1% de *.pdf*. Enfin, le dernier *repo* est celui dédié à notre site Internet, qui possède une double fonction : il nous permet de collaborer sur le site mais aussi de l'héberger grâce à *GitHub Pages*.

Ainsi, nous pouvons toujours être à jour sur la bonne version du jeu, du site ou des rapports, tout en étant géographiquement à distance les uns des autres.

5.2 Communication

Pour communiquer entre nous et avec notre équipe de β -testeurs, nous avons créé un serveur Discord³ découpé en multiples *channels*, ayant chacun une mission précise pour ne pas mélanger les informations. Ce serveur est aussi le lieu de nos réunions hebdomadaires (ou plus fréquemment en cas de soutenance). De plus, depuis le site Internet, nous avons mis un lien vers un questionnaire de remontée de bugs, via un *Google Forms*. Nous avons également mis en place un calendrier collectif lors de la dernière semaine avant la soutenance. Cela nous a offert une vision plus claire des tâches individuelles et de les découper en plages horaires afin de respecter les *deadlines* et de ne pas avoir d'erreurs de versions entre nous.

5.3 3D

Comme moteur de jeu, nous avons utilisé *Unity*. Comme Xavier utilise un Mac et que le reste du groupe est sous Windows, nous utilisons la version 2021.2.7f1 qui fonctionne sur les deux OS. Nous avons décidé de geler la version pour limiter au maximum les problèmes d'incompatibilité entre nous. Nous avons choisi *Unity* pour sa simplicité de prise en main et une fonctionnalité très utile : l'*Asset Store*. C'est une plateforme où nous pouvons acheter ou utiliser gratuitement des ressources telles que des bâtiments, des voitures ainsi que des personnages. En plus de l'utilisation de l'*Asset Store* pour notre première voiture, nous avons aussi utilisé *Blender*, logiciel de modélisation 3D. Il nous a permis entre autres de créer une Formule 1 de 2021 qui sera utilisée et implémentée plus tard dans le jeu.

5.4 IDE

Pour écrire notre code en C#, nous utilisons l'IDE *Rider* de *JetBrains*. Il possède une bonne intégration de Unity, et nous y sommes bien habitué, c'est

3. discord.gg/5NR43GHUBD

celui que nous utilisons au quotidien pour nos TPs de programmation. Pour faire certains tests, très précis et qui ne nécessitent pas de beaucoup de ressources, nous utilisons *Vim* directement dans notre terminal. Pour collaborer sur le rapport et les autres documents en L^AT_EX pour éviter de désigner un "esclave L^AT_EX", nous utilisons le site *Overleaf*⁴. À la manière d'un document en ligne, comme un *Google Docs* ou autre, nous pouvons écrire en simultané et compléter à quatre cerveaux les documents demandés.

5.5 Installateur

Pour pouvoir installer notre jeu sur Windows et Mac, il nous a fallu créer un fichier .exe permettant d'installer tous les fichiers nécessaires pour faire fonctionner *Bitume Legends*. Pour ce faire, nous avons utilisé *Inno Setup*, un logiciel libre et gratuit. Il nous a permis de donner un installateur aux β -testeurs plutôt que d'installer *Unity* et de télécharger le repository *GitHub*, et nous a également permis d'ajouter un fichier sauvegarde spécial pour certains β -testeurs qui nous ont demandé d'avoir un profil au niveau maximum pour pouvoir tester le jeu dans son intégralité. Ce logiciel nous permet également aux personnes allant sur notre site web de pouvoir télécharger notre jeu.

6 Notre ressenti au long du projet

6.1 Janvier \rightarrow Mars

6.1.1 Anthony

L'organisation du travail au sein du groupe était plutôt bonne. Dès le début, nous avons créé un repo GitHub afin de partager notre travail effectué ainsi qu'un Discord pour se mettre d'accord sur les différentes deadlines à propos du travail à effectué chaque semaine. Ainsi nous effectuons toutes les semaines, des réunions en vocal, dans le but d'expliquer aux autres membres du groupe ce que l'on a fait dans la semaine. A propos du site web, j'ai eu du mal à manipuler Bootstrap Studio du fait du manque de tutoriaux pour apprendre à utiliser Bootstrap Studio sans de développement en HTML. J'ai donc dû apprendre comment fonctionne ce logiciel et après avoir réalisé que le principe était de mettre tous les éléments liés dans une seule colonne, Bootstrap est devenu toute de suite très facile et m'a permis de réaliser rapidement le site web.

4. overleaf.com

6.1.2 Melvyn

Entre la première présentation du projet et le rapport de soutenance j'ai découvert les différentes difficultés liées à la réalisation d'un projet. J'ai eu du mal avec l'investissement personnel et la gestion du temps durant ce mois et demi. Je compte améliorer ma manière de travailler et changer mon processus de composition afin de pouvoir être plus investi dans le projet et produire plus de contenu. Je suis heureux de pouvoir travailler sur un style de jeu que j'aime. Je suis également ravi de pouvoir à nouveau travailler sur des projets musicaux et de pouvoir mettre ma passion au service du projet. Un aspect intéressant du projet a également été de pouvoir implémenter nos idées en utilisant des méthodes qui m'étaient dès lors inconnues. Ce fut un challenge mais j'ai trouvé cela amusant et suis prêt pour la suite.

6.1.3 Victorien

Quelques années auparavant, j'avais déjà créé un jeux-vidéo sur un autre moteur de jeux, Unreal Engine de Epic Games. Je n'avait cependant pas eu à coder car il suffisait de créer des Blueprints (éléments de code préfabriqués qu'il faut relier entre eux). De plus, c'était un jeu développé en solo, pas en équipe. Je ne savais pas spécialement par quoi commencer au vu de ce que l'on avait prévu. Après avoir fixé des objectifs hebdomadaires grâce aux réunions, les choses étaient cadrées et j'ai pu être plus efficace sur le développement du jeu. Il a fallu également faire attention à notre emploi du temps. Bien qu'il y eût les vacances pour avancer, il y avait à la rentrée les Midterms et 5 jours plus tard première soutenance. Nous devons être organisé et voir plus loin que la semaine suivante.

6.1.4 Xavier

La gestion du projet s'est très bien passée. Nous nous sommes rapidement mis d'accord sur une organisation qui nous convenait. Cela a permis d'avancer en restant tous sur la même longueur d'onde. Tout le groupe est très motivé, ce qui est nécessaire pour un projet de ce type. J'ai eu cependant beaucoup plus de mal à créer le mode multijoueur. J'ai dû m'y reprendre à trois fois en tout, car chaque version antérieure ne marchait pas ou n'acceptait pas les ajouts de fonctionnalités. Cela a occupé une bonne partie de mes soirées depuis mi-Janvier. Malgré cela, j'ai pu apprendre beaucoup de choses dans pleins de domaines, de l'hébergement Web en reprogrammant les DNS de GitHub, ou encore en recherchant comment créer une organisation GitHub pour gérer le projet.

6.2 Mars → Avril

6.2.1 Anthony

Au sujet du site Web, j'ai eu beaucoup de mal à implémenter le responsive design puisque l'application Bootstrap Studio 5 ne suffisait pas, et l'usage du code HTML était obligatoire. Ayant peu de connaissance dans ce langage, j'ai donc demandé de l'aide à mon groupe pour m'aider à le mettre en place. Ainsi après avoir compris le code généré par Bootstrap pour transformer chaque élément en responsive, d'autres problèmes de mise en forme ont complexifié la tâche entre les éléments responsive et non. De plus, la cohésion d'équipe a été fortement accentuée, ce qui a permis de régler les problèmes beaucoup plus rapidement. C'est pourquoi une grande avancée dans le jeu vidéo s'est faite ressentir par rapport à la soutenance 1. Du fait de cette avancée, ma motivation et l'envie de faire de Bitume Legends un bon jeu de voiture est encore plus forte que précédemment.

6.2.2 Melvyn

J'ai remarqué une légère baisse d'implication de ma part entre la première et la deuxième soutenance, notamment une semaine où le travail était plus que minime de ma part. J'ai également dû plus travailler avec les autres, comparé à la première soutenance où l'on travaillait un peu plus dans notre "domaine d'expertise" à chacun au lieu de nous entraider. Je me suis reprit quelques temps avant la soutenance et je suis fier de l'avancée du projet. Je pense que ce projet a beaucoup de potentiel. Il y a eu de grandes améliorations graphiques et techniques en ces quelques semaines et notre jeu ressemble enfin à un jeu. J'ai beaucoup d'espoirs pour la suite. Là où je m'étais trop concentré sur la musique lors de la première soutenance, je me suis plus tourné vers le graphisme, le gameplay et la physique du jeu, il était temps de vraiment faire du code...

6.2.3 Victorien

Suite à la première soutenance, la première idée que j'ai eu a été de vouloir développer le jeu et s'amuser dessus. Le but étant d'avoir un jeu plaisant, joli et agréable à jouer. J'ai donc passé de nombreuses heures à implémenter les différents modes de jeu, résoudre les bugs. Je suis très satisfait de mon travail. De plus, ceci m'a permis de m'améliorer en C# ainsi qu'en programmation orientée objet. C'est un vrai plaisir de coder le jeu et de voir notre travail porter ses fruits. Là où au départ Bitume Legends n'avait pas forcément de style et n'attirait pas l'œil, il est maintenant beaucoup plus attractif suite à la refonte graphique du jeu ainsi que notre avancée.

6.2.4 Xavier

Depuis la dernière soutenance, je me suis bien amusé à faire l'Intelligence Artificielle. Cela était sympa au début puis plus le temps avançait, plus les problèmes arrivaient. Cela m'a fait passé par tous les états possibles, de la joie intense à la dépression profonde. Malgré cela, l'Intelligence Artificielle a sûrement été la partie que j'ai préféré faire. Pour le reste, je suis très fier de l'avancée que nous avons, nous sommes à jour sur notre planning et le jeu est très plaisant à jouer. Nous sommes très content de ce que rendent les graphiques et les voitures, ce qui était le point noir de la dernière soutenance. Bref, le jeu va vraiment être super sympa et cela nous rend heureux !

6.3 Avril → Juin

6.3.1 Anthony

Ce projet m'a servi à découvrir le travail de groupe sur une assez longue période, avec tous ses inconvénients et ses avantages. Il m'a été très enrichissant, vis à vis de régler les bugs en cherchant des solutions le plus vite possible. De plus, j'ai ressenti un vrai travail d'entraide entre nous mais aussi en terme de répartition des tâches ce qui a considérablement mis un boost dans l'avancer de notre jeu. Au final, je suis plutôt très content du résultat du jeu, nous avons été optimiste au début du jeu sur ce qu'on voulait à la fin et nous avons pratiquement rempli tous nos objectifs.

6.3.2 Melvyn

Suite à la deuxième soutenance, je me suis mieux investi dans le projet, notamment dans le travail de l'Intelligence Artificielle. Je me suis amusé à faire de l'optimisation du système et le rendre compétitif. Je me suis vraiment épanoui dans le projet à partir du moment où nous avons véritablement travaillé tous ensemble. Ce projet a été très enrichissant à titre personnel et également professionnellement. C'est le premier pavé sur la route des futurs travaux de groupes en tant qu'ingénieur. Bien que le fait que ce projet devait être travaillé sur notre temps personnel, je n'ai pas eu l'impression d'avoir manqué ce temps. Je me réjouis de travailler à nouveau sur un projet d'une telle envergure, surtout avec l'expérience acquise ce semestre.

6.3.3 Victorien

Ce projet a été très enrichissant. Il m'a permis de comprendre certains concepts que je n'avais pas spécialement compris en C#, mais également sur le plan humain

où il nous a fallu rester soudé et donc trouver des moyens de rester motivé ou d'être motivé, et de rester une équipe. J'ai pu également remarqué que l'on était bien plus efficace lorsque nous étions tous en présentiel à quatre autour d'une table et l'ambiance était bien plus plaisante, ce qui sera utile lors des futurs projets. Poser également les bases avec des réunion et une manière de travailler en début de projet nous a également été favorable et nous a permis d'être plus efficace rapidement. Les synthèses à la fin de chaque soutenance nous ont également permis de savoir sur quoi accés nos prochains objectif. Cette méthode nous sera, comme dit plus, certainement tous utile lors de nos prochains projet. Enfin, je suis très contents de voir ce projet aboutir, de savoir que nous sommes parti de rien et que nous avons un jeu fonctionnel et sur lequel chacun des membres des groupes mais également les joueurs se plaisent.

6.3.4 Xavier

Je suis très content de notre projet. Nous avons su aller jusqu'au bout de nos idées phares, fait le tri qui s'imposait parmi les non essentielles, et créé un jeu qui nous donne vraiment envie de jouer. Malgré les contretemps, et les soucis d'organisations, propres à chaque travail de groupe, nous avons pu avancer selon notre timeline définie en début de projet, dans notre cahier des charges.

7 Ressentis de certains testeurs

Le programme de β -testing a permis que chaque utilisateurs, ou testeurs, puissent nous remonter les bugs éventuels lorsqu'ils jouent au jeu. De plus, ce dernier sert à remonter ce que le joueur à le plus aimer, le moins et même ce qu'il aimerait voir. Ce programme à démontrer une très grande efficacité dans l'originalité du jeu, notamment grâce aux remontées d'idées des testeurs mais nous a permis aussi d'avoir un jeu très solide, avec peu de bugs à notre connaissance grâce aux nombreux testeurs. Cependant, deux endroits étaient réservés pour le programme de β -testing, un sur le site Internet avec le Google Form et un autre sur notre discord. Nous avons ainsi remarqué que les testeurs ont pour, quasi de la totalité, utilisés le serveur discord. Maintenant, nous avons décidé de vous transmettre quelque retours, avis du jeu de certains testeurs.

8 Problèmes rencontrés et solutions

8.1 Janvier → Avril

8.1.1 Implémentation des voitures

L'implémentation de la physique de voitures fut complexe, en particulier au vu des nombreuses variables impactant le comportement d'une voiture. Il a fallu gérer le poids, la vitesse, la puissance en chevaux du moteur, le couple maximum, l'angle et la vitesse de braquage pour la direction, la force et la vitesse de freinage ou encore l'inertie du moteur et la hauteur du centre de gravité.

Suite à l'implémentation de la physique des voitures, nous nous sommes rendu compte que les modèles des voitures que nous avions ne permettaient pas son bon fonctionnement. Il a fallu dans un premier temps modifier les *prefabs* des voitures, entre autres leurs *rigibody* qui empêchaient les roues de tourner et de considérer qu'elles touchaient le sol. Suite à cela, il a fallu également modifier les roues des voitures en unifiant la jante ainsi que le pneu, chose qui n'était pas faite, avant pour pouvoir entraîner l'essieu et synchroniser la direction. Après avoir résolu ce problème sur une voiture, il a fallu appliquer la solution aux autres, ce qui explique également pourquoi les voitures ont pour l'instant la même physique. Toutefois, les voitures ainsi que les collisions sont fonctionnelles.

Un autre problème fût aussi celui du sound-design de la voiture. Plusieurs éléments dans une voiture produisent du son et n'ont pas le même comportement en fonction du poids et la vitesse du véhicule ou encore de la puissance du moteur. Il a fallu gérer le passage des rapports, le bruit du moteur dont le *pitch* et le volume changeait selon la vitesse ou le type du véhicule (un pick-up a un bruit différent d'une Supercar). Avec le passage de rapport il y avait également le bruit du *turbocompresseur* ou du *supercharger* à gérer.

Au final nous avons trouvé certains paramètres permettant à la voiture d'avoir un bon comportement et d'avoir un bon son. Il ne nous reste plus qu'à appliquer ces paramètres aux autres voitures.

8.1.2 Intelligence Artificielle

En implémentant notre IA, nous avons fait face à de nombreux problèmes divers et variés. Pour commencer, les voitures autonomes ne voulaient pas passer par les bons endroits. Elles faisaient demi-tour sur la ligne de départ et passaient à travers les murs ou entre les plots et se stoppaient sans raison au milieu du circuit. Pour résoudre ce problème, nous avons compris comment fonctionnait le *NavMesh* puis appris à séparer les différentes couches de circuit : ce sur quoi la voiture peut rouler et les obstacles. Le *NavMesh* est un calque qui est apposé au dessus du circuit et qui permet de définir une zone sur laquelle les voitures autonomes peuvent

rouler et calculer leur itinéraire. Après cela, nous avons recalculé le *NavMesh* et remis des plots ou des barrières là où c'était nécessaire.

Le deuxième problème consistait en un arrêt inattendu et systématique au milieu du circuit. Pour trouver comment résoudre ce problème, nous avons exploré plusieurs pistes. Premièrement vérifier si la direction de la voiture était la bonne, puis si les *checkpoints* fonctionnaient correctement. Après de longues recherches infructueuses, nous avons trouvé que le problème était lié à l'absence de composants dans la voiture, permettant de comptabiliser les collisions avec les *checkpoints*. Nous avons donc rajouté ce composant et changé la façon de définir une trajectoire. Et après ce travail, les voitures autonomes arrivent à la fin du circuit.

8.1.3 Implémentation des musiques

Pour les menus nous avons préféré créer plusieurs scènes pour les sous-menus accessibles par le menu principal., plutôt qu'une seule scène qui se transforme à l'appui d'un bouton sur le menu. Bien que pratique pour les scripts individuels à chaque modes de jeux et sous-menus, cela compliquait l'implémentation de la musique. Il a fallu trouver un moyen d'empêcher les musiques de s'arrêter à chaque chargement d'un nouveau menu. Pour cela il fallait faire en sorte que la musique fasse partie des éléments *DontDestroyOnLoad*, qui permet qu'une fois la musique chargé, elle restera possible d'accès même après changement de scène. Il nous a ensuite fallu dans le script s'occupant de la musique, de lui faire référence et de lui dire de ne pas s'arrêter lors d'un changement de scène.

Un problème qui a suivi est le fait que bien que la musique ne s'arrête plus, à chaque ouverture d'un menu une autre instance de la musique se lançait. Pour remédier à cela il a fallut préciser dans le script que toute nouvelles musique dans ne scène de menu ne devait pas se lancer si une autre se jouait déjà.

Enfin un dernier problème a eu lieu lors de l'implémentation des musiques de courses. La musique de menu continuait de jouer par dessus celle de course. Il a alors fallut ajouter dans le script le cas où lorsque l'on était en course, il fallait mettre la propriété *SetActive*, qui permet d'activer ou désactiver un élément. Par exemple, si l'on entrait en course, il a fallu désactiver la musique de menu et activer la musique de course.

8.2 Avril → Juin

8.2.1 Implémentation des voitures

L'implémentation de la physique de voitures fut complexe, en particulier au vu des nombreuses variables impactant le comportement d'une voiture. Il a fallut gérer le poids, la vitesse, la puissance en chevaux du moteur, le couple maximum,

l'angle et la vitesse de braquage pour la direction, la force et la vitesse de freinage ou encore l'inertie du moteur et la hauteur du centre de gravité.

Suite à l'implémentation de la physique des voitures, nous nous sommes rendu compte que les modèles des voitures que nous avions ne permettaient pas son bon fonctionnement. Il a fallu dans un premier temps modifier les *prefabs* des voitures, entre autres leurs *rigibody* qui empêchaient les roues de tourner et de considérer qu'elles touchaient le sol. Suite à cela, il a fallu également modifier les roues des voitures en unifiant la jante ainsi que le pneu, chose qui n'était pas faite, avant pour pouvoir entraîner l'essieu et synchroniser la direction. Après avoir résolu ce problème sur une voiture, il a fallu appliquer la solution aux autres, ce qui explique également pourquoi les voitures ont pour l'instant la même physique. Toutefois, les voitures ainsi que les collisions sont fonctionnelles.

Un autre problème fût aussi celui du sound-design de la voiture. Plusieurs éléments dans une voiture produisent du son et n'ont pas le même comportement en fonction du poids et la vitesse du véhicule ou encore de la puissance du moteur. Il a fallu gérer le passage des rapports, le bruit du moteur dont le *pitch* et le volume changeait selon la vitesse ou le type du véhicule (un pick-up a un bruit différent d'une Supercar). Avec le passage de rapport il y avait également le bruit du *turbocompresseur* ou du *supercharger* à gérer.

Au final nous avons trouvé certains paramètres permettant à la voiture d'avoir un bon comportement et d'avoir un bon son. Il ne nous reste plus qu'à appliquer ces paramètres aux autres voitures.

8.2.2 Implémentation des musiques

Pour les menus nous avons préféré créer plusieurs scènes pour les sous-menus accessibles par le menu principal., plutôt qu'une seule scène qui se transforme à l'appui d'un bouton sur le menu. Bien que pratique pour les scripts individuels à chaque modes de jeux et sous-menus, cela compliquait l'implémentation de la musique. Il a fallu trouver un moyen d'empêcher les musiques de s'arrêter à chaque chargement d'un nouveau menu. Pour cela il fallait faire en sorte que la musique fasse partie des éléments *DontDestroyOnLoad*, qui permet qu'une fois la musique chargé, elle restera possible d'accès même après changement de scène. Il nous a ensuite fallu dans le script s'occupant de la musique, de lui faire référence et de lui dire de ne pas s'arrêter lors d'un changement de scène.

Un problème qui a suivi est le fait que bien que la musique ne s'arrête plus, à chaque ouverture d'un menu une autre instance de la musique se lançait. Pour remédier à cela il a fallu préciser dans le script que toute nouvelles musique dans ne scène de menu ne devait pas se lancer si une autre se jouait déjà.

Enfin un dernier problème a eu lieu lors de l'implémentation des musiques de courses. La musique de menu continuait de jouer par dessus celle de course. Il a

alors fallut ajouter dans le script le cas où lorsque l'on était en course, il fallait mettre la propriété *SetActive*, qui permet d'activer ou désactiver un élément. Par exemple, si l'on entrait en course, il a fallu désactiver la musique de menu et activer la musique de course.

8.2.3 Intelligence Artificielle

En implémentant notre Intelligence Artificielle, nous avons fait face à de nombreux problèmes divers et variés. Pour commencer, les voitures autonomes ne voulaient pas passer par les bons endroits. Elles faisaient demi-tour sur la ligne de départ et passaient à travers les murs ou entre les plots et se stoppaient sans raison au milieu du circuit. Pour résoudre ce problème, nous avons compris comment fonctionnait le *NavMesh* puis appris à séparer les différentes couches de circuit : ce sur quoi la voiture peut rouler et les obstacles. Le *NavMesh* est un calque qui est apposé au dessus du circuit et qui permet de définir une zone sur laquelle les voitures autonomes peuvent rouler et calculer leur itinéraire. Après cela, nous avons recalculé le *NavMesh* et remis des plots ou des barrières là où c'était nécessaire. Le deuxième problème consistait en un arrêt inattendu et systématique au milieu du circuit. Pour trouver comment résoudre ce problème, nous avons exploré plusieurs pistes. Premièrement vérifier si la direction de la voiture était la bonne, puis si les *checkpoints* fonctionnaient correctement. Après de longues recherches infructueuses, nous avons trouvé que le problème était lié à l'absence de composants dans la voiture, permettant de comptabiliser les collisions avec les *checkpoints*. Nous avons donc rajouté ce composant et changé la façon de définir une trajectoire. Et après ce travail, les voitures autonomes arrivaient à la fin du circuit.

8.2.4 Multijoueur

TODO XAVIER Donec tempus neque vitae est. Aenean egestas odio sed risus ullamcorper ullamcorper. Sed in nulla a tortor tincidunt egestas. Nam sapien tortor, elementum sit amet, aliquam in, porttitor faucibus, enim. Nullam congue suscipit nibh. Quisque convallis. Praesent arcu nibh, vehicula eget, accumsan eu, tincidunt a, nibh. Suspendisse vulputate, tortor quis adipiscing viverra, lacus nibh dignissim tellus, eu suscipit risus ante fringilla diam. Quisque a libero vel pede imperdiet aliquet. Pellentesque nunc nibh, eleifend a, consequat consequat, hendrerit nec, diam. Sed urna. Maecenas laoreet eleifend neque. Vivamus purus odio, eleifend non, iaculis a, ultrices sit amet, urna. Mauris faucibus odio vitae risus. In nisl. Praesent purus. Integer iaculis, sem eu egestas lacinia, lacus pede scelerisque augue, in ullamcorper dolor eros ac lacus. Nunc in libero.

Fusce suscipit cursus sem. Vivamus risus mi, egestas ac, imperdiet varius, faucibus quis, leo. Aenean tincidunt. Donec suscipit. Cras id justo quis nibh scelerisque

dignissim. Aliquam sagittis elementum dolor. Aenean consectetur justo in pede. Curabitur ullamcorper ligula nec orci. Aliquam purus turpis, aliquam id, ornare vitae, porttitor non, wisi. Maecenas luctus porta lorem. Donec vitae ligula eu ante pretium varius. Proin tortor metus, convallis et, hendrerit non, scelerisque in, urna. Cras quis libero eu ligula bibendum tempor. Vivamus tellus quam, malesuada eu, tempus sed, tempor sed, velit. Donec lacinia auctor libero.

Conclusion du projet

8.3 Objectif atteint ?

Par nécessité nous avons créé un compte Instagram pour notre groupe afin d'informer et de discuter avec la communauté sur les avancements du jeu. Cependant notre serveur *Discord* a prit une certaine ampleur et la majorité des annonces et des discussions vis-à-vis du jeu avec nos *Beta-testeurs*. Nous avons donc choisis d'abandonner l'idée de maintenir le compte *Instagram* aux profits de *Discord*.

Lors de la deuxième soutenance, une remarque a été faite concernant notre trop grand nombre de voitures, ce qui nous a amené à revoir nos objectifs, notamment vis-à-vis de la customisation des véhicules. L'optimisation de chaque véhicule même lorsque leur nombre était réduit nous a prit déjà un temps considérable, alors si l'on devait permettre une modification des statistiques du véhicule tout en les gardant fonctionnelles et équilibrées, tout le reste du projet aurait été impacté. De plus, un système de customisation aurait impliqué un système de gain d'argent virtuel pour acheter ces modifications en plus du gain d'expérience. Finalement nous nous sommes concentrés sur la poignée de véhicules déblocables afin de leur donner une certaine âme et une identité via leur caractéristiques et leur son.

Un nombre d'une demi-douzaine de musique avait également été prévue à l'origine, encore une fois le problème venait du temps que cela aurait nécessité par rapport aux fonctionnalités importantes du jeu. Si l'on se réfère à notre "RAPPORT TEMPS PROJET"

8.4 Avenir de Bitume Legends

+ projet Hopital à la toussaint 2022

Entre la première et la deuxième soutenance notre jeu a gagné en popularité au sein des autres groupes et même au sein de l'école. En effet la refonte graphique du jeu ainsi que la création des modes de jeu *Solo* et *Timer* a plu à certains qui aiment également les jeux de voiture. De plus le lancement de la *Beta* ouverte a fait découvrir le projet zen dehors de l'école. Les retours étaient globalement très positifs et nous permettaient une découverte et une résolution plus facile des bugs.

En restant proche de nos testeurs nous avons pu améliorer la qualité du jeu et développer une communauté de passionnés qui subsiste en dehors de l'école.

8.5 Remerciements

Nous remercions les membres de notre communauté de β -testeurs, qui nous ont beaucoup aidé dans la correction de nombreux *bugs*. Nous remercions pareillement Mace24 qui nous a conseillé tout au long du projet sur la gestion de notre temps et le déroulement des soutenances, ainsi qu'un rapport de *bugs* très complet. Nous remercions également les personnes qui ont soutenus et cru en notre projet. Merci aux personnes qui e sont dévoués pour nous permettre de compléter notre procédure d'installation !

Made with ♡ by CARRENIX on L^AT_EX.
© 2021-2022, *Bitume Legends*
btms.games

Références

- blender.org
- bootstrapstudio.io
- discord.com
- unity.com
- photonengine.com/pun
- overleaf.com
- jetbrains.com/rider
- assetstore.unity.com
- [image – line.com/fl – studio](https://image-line.com/fl-studio)
- youtube.com
- github.com

Liste des tableaux

1	Tableau récapitulatif de notre projet	5
2	Chronologie du Projet	30

Table des figures

TABLE 2 – Chronologie du Projet

Before	•	To Complete
10 mars	•	Soutenance 1
14 mars	•	Mise en place des objectifs pour la soutenance 2
20 mars	•	Implémentation du drift et du sound design
25 mars	•	Création de deux nouveaux circuits
30 mars	•	Implémentation des menus <i>Timer</i> et <i>Solo</i>
10 avril	•	Intelligence Artificielle version 1.0
15 avril	•	Implémentation du début et de la fin des courses
17 avril	•	Implémentation du garage
18 avril	•	Implémentation de la sauvegarde
20 avril	•	Intelligence Artificielle version 2.0
22 avril	•	Finition du menu principal
24 avril	•	Implémentation de la musique
25 avril	•	Mode Timer fonctionnel
27 avril	•	Site Internet responsive
28 avril	•	Mode Solo fonctionnel
29 avril	•	Soutenance 2
After	•	To Complete